



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Monolitické a iterační řešení Biotovy poroelasticity

Bakalářská práce

Studijní program: B3901 – Aplikované vědy v inženýrství
Studijní obor: 3901R055 – Aplikované vědy v inženýrství
Autor práce: **Sabina Bednářová**
Vedoucí práce: Mgr. Jan Stebel, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

Monolithic and iterative solution of Biot's poroelasticity

Bachelor thesis

Study programme: B3901 – Applied Sciences in Engineering
Study branch: 3901R055 – Applied Sciences in Engineering
Author: **Sabina Bednářová**
Supervisor: Mgr. Jan Stebel, Ph.D.





Zadání bakalářské práce

Monolitické a iterační řešení Biotovy poroelastivity

Jméno a příjmení: **Sabina Bednářová**
Osobní číslo: M16000097
Studijní program: B3901 Aplikované vědy v inženýrství
Studijní obor: Aplikované vědy v inženýrství
Zadávací katedra: Ústav nových technologií a aplikované informatiky
Akademický rok: **2018/2019**

Zásady pro vypracování:

1. Seznamte se:
 - (i) s Biotovou teorií poroelastivity a jejím využitím pro modelování hydrogeologických procesů,
 - (ii) s iteračními schémata a dimenzionální redukcí pro Biotův systém,
 - (iii) s výpočetní knihovnou FEniCS a základními metodami pro řešení soustav lineárních rovnic.
2. Aplikujte iterační schéma "fixed-stress splitting" na redukovaný Biotův systém. Implementujte toto schéma do existujícího výpočetního kódu pro redukovaný Biotův systém ve 2D. Provedte výpočetní analýzu konvergence.
3. Na základě předchozích zkušeností navrhnete vhodnou volbu relaxačních parametrů pro oblast a puklinu v závislosti na parametrech modelu, zajišťující konvergenci iteračního schématu.
4. Porovnejte výpočetní náročnost monolitického a iteračního řešení redukovaného Biotova systému v závislosti na počtu elementů sítě a fyzikálních parametrech.

Rozsah grafických prací: dle potřeby
Rozsah pracovní zprávy: 30 – 40 stran
Forma zpracování práce: tištěná/elektronická



Seznam odborné literatury:

- [1] CHENG, A. H.-D. Poroelasticity. Springer, 2016.
- [2] BOTH, J.W., BORREGALES, M., NORDBOTTEN, J.M., KUMAR, K., RADU, F.A. Robust fixed stress splitting for Biot's equations in heterogeneous media, Appl Math Lett 68:101-108, 2017.
- [3] ANGOT, P., BOYER, F., HUBERT, F. Asymptotic and numerical modelling of flows in fractured porous media, ESAIM:M2AN 43:239-275, 2009.
- [4] TEBBENS, J.D., HNĚTYNKOVÁ, I., PLEŠINGER, M., STRAKOŠ, Z., TICHÝ, P. Analýza metod pro maticové výpočty Základní metody. MatfyzPress, 2012.
- [5] FEniCS Project [online, cit. 24. 9. 2018]. Dostupné z <https://fenicsproject.org>.

Vedoucí práce: Mgr. Jan Stebel, Ph.D.
Ústav nových technologií a aplikované informatiky
Datum zadání práce: 18. října 2018
Předpokládaný termín odevzdání: 30. dubna 2019

L. S.

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

Ing. Josef Novák, Ph.D.
vedoucí ústavu

V Liberci 18. října 2018

Prohlášení

Byla jsem seznámena s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědoma povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracovala samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že texty tištěné verze práce a elektronické verze práce vložené do IS STAG se shodují.

30. 4. 2019

Sabina Bednářová

Poděkování

V první řadě bych ráda poděkovala vedoucímu své práce, panu Mgr. Janu Steblovi, Ph.D., za neskutečnou trpělivost, ochotu a velké množství času, které mi věnoval, za pomoc s mojí češtinou a skvělé vedení, které mě donutilo na práci intenzivně pracovat po celý školní rok.

Dále bych ráda poděkovala doc. Ing. Janu Šemberovi, Ph.D., za ochotnou diskuzi po vyučovacích hodinách a za dodání kuráže ve chvílích, kdy mi bylo spíše do breku.

Děkuji také svojí rodině, bez jejíž podpory bych se nikdy takhle daleko nedostala, hlavně svým rodičům, že mi šli v tomhle směru příkladem.

Abstrakt

Tato bakalářská práce se zabývá numerickým řešením Biotova systému v prostředí s puklinami. Je odvozeno iterační schéma typu fixed-stress splitting a popsána jeho implementace s využitím knihovny FEniCS. Dále je provedena výpočetní analýza konvergence v závislosti na relaxačních parametrech a na fyzikálních parametrech. Na základě předchozích zkušeností je naržena volba relaxačních parametrů pro oblast a puklinu, zajišťující konvergenci iteračního schématu. Nakonec je porovnána výpočetní náročnost monolitického a iteračního řešení redukovaného Biotova systému v závislosti na počtu elementů sítě a fyzikálních parametrech.

Klíčová slova: poroelasticita, Biotův redukovaný model, fixed-stress splitting, monolitické řešení, iterační řešení

Abstract

This bachelor thesis deals with numerical solution of Biot's system in domains containing fractures. An iterative fixed-stress type splitting is derived and its implementation using FEniCS library is described. Further we perform computational convergence analysis according to the dependency on tuning parameters and physical parameters. Based on previous experience, we suggest a convenient choice of tuning parameters for the domain and the fracture, ensuring convergence of iterative scheme and compare computational costs of monolithic and iterative solution depending on the quantity of mesh elements and physical parameters.

Keywords: poroelasticity, Biot's reduced model, fixed-stress splitting, monolithic solution, iterative solution

Obsah

1	Úvod	8
1.1	Porézní materiály	8
1.2	Modelování hydraulického štěpení	9
1.3	Biotův model a jeho varianta pro prostředí s puklinami	10
2	Přehled numerických metod pro klasický Biotův model	12
2.1	Monolitické (implicitní) řešení - přímé a iterační metody	12
2.2	Iterační (sekvenčně implicitní) řešení	14
3	Numerické řešení Biotova modelu pro oblast s puklinou	16
3.1	Monolitické řešení	21
3.2	Aplikace fixed stress splitting	23
3.3	Analýza konvergence	28
4	Závěr	39

1 Úvod

V dnešní době, kdy řešíme spoustu environmentálních problémů, se snažíme zapojovat více ekologická řešení, jako je například obnovitelná energie. Obnovitelná energie je taková, která se v lidském časovém měřítku přirozeně obnovuje, na rozdíl od neobnovitelné energie, jako jsou třeba fosilní paliva, které se v lidském časovém měřítku neobnovují a tak může dojít k jejich vyčerpání, nebo náklady na těžení těchto paliv budou tak vysoké, že je už nebude ekonomicky vůbec výhodné používat. Mimo to většina neobnovitelné energie negativně přispívá ke změně klimatu.

Jedním z typů obnovitelné energie je geotermální energie, kdy využíváme tepelný projev zemského jádra. Toto využití geotermální energie je ne vždy obnovitelné - některé zdroje geotermální energie lze vyčerpat v horizontu desítek let, a velkou nevýhodou je také velká závislost na geografickém umístění.

V systému Hot Dry Rock (HDR) jde o využití akumulovaného tepla v hlubinné zóně s vysokou propustností. Takovou zónu je obvykle nutné uměle vytvořit, například s využitím existující sítě puklin. Pomocí Biotovy teorie poroelasticity chceme sledovat, studovat proces otevírání pro využití geotermální energie.

Biotova poroelasticita je soustava rovnic, která popisuje interakci mezi prouděním a deformací porézního média. Z takové interakce můžeme zjistit, jak se bude puklina v tomto médiu otevírat či zavírat, pokud do ní budeme vtlačet kapalinu pod velmi vysokým tlakem.

Cílem této bakalářské práce je seznámení se s řešením Biotových rovnic poroelasticity a porovnání náročnosti dvou metod řešení: monolitické a iterační. K porovnání těchto dvou řešení je použita výpočetní knihovna FeniCS.

1.1 Porézní materiály

Celý náš okolní svět je obklopen porézními materiály, některé z nich jsou přírodní, jiné jsou uměle vyrobené člověkem. Existuje jich mnoho v různých formách - patří sem granulované materiály (např. písek), kde každá částice křemenu má jinou

velikost a částice nejsou pevně spojené, proto jimi může tekutina volně protékat bez většího odporu - toto jsou takzvané granulované materiály. Naopak například pískovec, což jsou částice písku pevně spojené kalcitem a jílem, je porézní pevnou látkou. Přestože jsou částice pískovce k sobě pevně spojeny a tekutina nemůže skrz pískovec protékat tak volně jako hromadou písku, může být obecně porézní pevná látka dobrým vodivým médiem. V každém porézním médiu se přirozeně vyskytují póry, puklinky nebo trhliny, v nichž se může kapalina šířit. Velikost pórů se obecně v každém porézním médiu liší, v závislosti na tom, jak vznikalo. Například vulkanická hornina má póry o velikosti až jednoho centimetru - při chlazení lávy se vytvořily plynové bublinky z rozpouštění těkavých látek [1]. Další porézní horninou je například žula, která je krystalickou horninou a pro HDR systém je žulový masiv vhodný. Schopnost porézního materiálu propouštět vodu se nazývá permeabilita (propustnost).

Většina přírodně vyskytujících se materiálů je porézních. I lidské tělo je porézním materiálem. Mezi uměle vytvořené porézní materiály patří například beton (tato látka má relativně malou propustnost) nebo různé stavařské pěny.

1.2 Modelování hydraulického štěpení

Hydraulické štěpení je metoda pro tvorbu puklin v horninovém masivu. Tato metoda není pro náš svět ničím novým; první aplikace proběhly již v roce 1947 [2]. Jde o vytvoření hlubinného vrtu, do kterého se poté vtlačí kapalina pod velmi vysokým tlakem. Pokud tlak kapaliny překoná pevnost horniny, dojde k jejímu rozštěpení a tvoření puklin, kterými se poté kapalina dále šíří (nejčastěji horizontálně) a tím zvětšuje puklinu [3].

Při tvoření puklin mohou nastat různé případy - pukliny různé šířky, různého směru a délky, ale jen určité pukliny jsou použitelné pro konkrétní průmyslové aplikace. Tvoření puklin je možné částečně ovlivnit, například do určitého směru či typu pukliny. Ovlivnění tvorby probíhá změnou tlaku, teploty nebo změnou kapaliny, která je použita pro tvorbu puklin [1].

Systémy HDR vyžadují, aby horniny byly v hloubce rozpraskány. Tvoří se (minimálně) dva vrty, do kterých se vtlačí kapalina, která vytvoří pukliny v okolí konce vrtu. Tyto dva vrty jsou k sobě dostatečně blízko, tak, aby vytvořené pukliny byly propojené a mohly mezi sebou komunikovat. Do jednoho vrtu se čerpá voda, která se ohřeje od okolní horniny a pak se vrátí na povrch druhým vrtem.

Modelování hydraulického štěpení je žádoucí a výhodné v případech, kdy chceme

například předejít nežádoucí seismicitě nebo ovlivnit vlastnosti propustné zóny.

1.3 Biotův model a jeho varianta pro prostředí s puklinami

Maurice Anthony Biot (1905-1985) byl belgicko-americký aplikovaný fyzik, který se vedle příspěvků v oblasti termodynamiky, letectví, zemětřesení, geofyziky a elektromagnetismu, považuje za zakladatele teorie poroelasticity.

Prvním krokem k moderní teorii poroelasticity bylo představení obecné trojrozměrné teorie pružné deformace v nasyceném tuhém mediu [4]. Tato teorie však nebyla ještě úplně dokonalá, neboť byla omezena na izotropní případ. V 50. letech minulého století M. A. Biot svou teorii rozšířil na anizotropní případy a odvodil rovnice pro dynamickou odezvu porézních médií [5], [6]. Přestože tato forma Biotovy teorie již získala mnoho úspěchů a využití, stále šlo pouze o teorii s předpokladem lineární elasticity. Až o několik let později, v 70. letech 20. století, byla tato teorie znovu upravena a zdokonalena pro obecnější případy, kdy M. A. Biot zahrnul i možnost nelineární elasticity [7].

Základní myšlenky teorie poroelasticity spočívají v tom, že tlak kapaliny přispívá k celkovému namáhání v porézním mediu a samotný tlak kapaliny může napnout porézní medium. V takovémto pórovitém mediu dochází k pohybu kapaliny kvůli rozdílům tlaku kapaliny v pórech, které souvisí s mechanickým zatížením porézního média. V důsledku toho může dojít k posunu, otvírání nebo zavírání puklin.

Nelineární chování je obvykle asociováno se vznikem a šířením puklin [1]. Pro náš případ uvažujeme médium, které má pevný skelet (dále označované jako matrice). Předpokládáme, že kapalina je nestlačitelná a že porézní medium stlačitelné je. Biotova teorie pracuje s předpokladem rozkladu vnitřního napětí v plně saturovaném mediu. Toto napětí lze rozdělit na dvě části: rovnováha sil v hornině a zákon zachování hmotnosti v tekutině.

Rovnováha sil v hornině popisuje změnu tvaru media (deformace matrice) a je popsána touto rovnicí:

$$-\nabla \cdot \boldsymbol{\sigma} + \alpha \nabla p = \mathbf{f}, \quad (1.1)$$

kde $\boldsymbol{\sigma}$ představuje tenzor napětí, α je Biotův koeficient, p popisuje tlak v pórech a \mathbf{f} představuje objemovou sílu.

Materiálový zákon zahrnutý v Biotově teorii se nazývá Hookův zákon:

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\epsilon}(\mathbf{u}) + \lambda(\boldsymbol{\nabla} \cdot \mathbf{u})\mathbb{I}. \quad (1.2)$$

Zde μ, λ jsou Lamého parametry, $\boldsymbol{\epsilon}(\mathbf{u})$ je tenzor malých deformací, \mathbf{u} je vektor posunutí a \mathbb{I} je jednotková matice. Zákon zachování hmotnosti v tekutině je popsán rovnicí:

$$\partial_t(Sp + \alpha \boldsymbol{\nabla} \cdot \mathbf{u}) - \boldsymbol{\nabla} \cdot \mathbf{q} = g, \quad (1.3)$$

kde první člen ve tvaru časové derivace je přírůstek objemu kapaliny za jednotku času. Vektor \mathbf{q} se nazývá Darcyho tok nebo Darcyho rychlost a popisuje výtok na jednotku plochy. S značí storativitu (někdy značeno také jako $1/M$) a g představuje zdroj objemové síly.

Darcyho zákon pro tok tekutin porézním médiem představuje rovnice:

$$\mathbf{q} = -\kappa \boldsymbol{\nabla} p, \quad (1.4)$$

kde κ je hydraulická vodivost media, μ popisuje viskozitu média, $\boldsymbol{\nabla} p$ popisuje celkový pokles tlaku.

Rovnice (1.1) - (1.4) jsou známé jako tzv. Biotův systém pro deformovatelné porézní médium. Pro prostředí obsahující pukliny byl pomocí tzv. dimenzionální redukce odvozen Biotův systém, který bude popsán v kap. 3 a který je hlavním předmětem studia této práce.

2 Přehled numerických metod pro klasický Biotův model

Jako téměř všechny parciální diferenciální rovnice (PDR), i Biotův systém nemá (až na několik specifických případů) analytické řešení, proto se většinou musí vyřešit numericky.

Numerické metody, jako např. metoda konečných prvků, aproximují PDR soustavu algebraických rovnic. Pro numerické řešení Biotových rovnic jsou v dnešní době používány především dva přístupy: plně implicitní (monolitický) a sekvenčně implicitní (iterační). Plně implicitní přístup vyžaduje řešení celé soustavy rovnic současně, což přináší výhodu bezpodmínečné stability, ale vyžaduje robustní metody a předpomiňovače.

Sekvenčně-implicitní přístup nabízí větší flexibilitu v návrhu kódu a použití specializovaných metod pro řešení dílčích problémů, než plně implicitní přístup [8].

2.1 Monolitické (implicitní) řešení - přímé a iterační metody

Numerické řešení rovnic (1.1) - (1.4) představuje výzvu, protože rovnice pružnosti a proudění jsou plně svázány a vzniklá matice má složitou strukturu. Pro monolitické řešení se používají dva základní typy metod - přímé a iterační.

Uvažujeme soustavu

$$\mathbb{A}\mathbf{x} = \mathbf{b} \quad (2.1)$$

s maticí $\mathbb{A} \in \mathbb{R}^{n \times n}$ a vektorem pravých stran $\mathbf{b} \in \mathbb{R}^n$. Nejznámější přímou metodou je řídká LU dekompozice (Gaussova eliminace) pro řešení lineární soustavy rovnic. Její maticová reprezentace je

$$\mathbb{A} = \mathbb{L}\mathbb{U}, \quad (2.2)$$

kde \mathbb{L} a \mathbb{U} jsou trojúhelníkové (tedy snadno invertovatelné) matice. Tato metoda

je robustní, velmi jednoduchá a použitelná obecně pro regulární matice. Je to doporučená metoda pro systémy s až několika tisíci neznámými (v praxi i více, pokud je matice řídká) a proto může být dobrou volbou pro mnoho 2D problémů a menší 3D problémy. Pro větší problémy se řídký LU rozklad zpomaluje a rychle se vyčerpá paměť, protože řídká matice se rychle zaplní [9]. Cena klasické Gaussovy eliminace a Gaussovy eliminace s pivotací měřená počtem aritmetických operací pro regulární matici je stejná. Přímý chod Gaussovy eliminace vyžaduje pro matici řádu $n \times n$ až $O(n^3)$, viz tabulku 2.1 [10]. Pro velké problémy jsou dobrou alternativou iterační metody, které jsou flexibilnější a vyžadují o dost méně paměti.

Tabulka 2.1: Počty jednotlivých operací přímého chodu Gaussovy eliminace.

Operace	celkem
\times	$\frac{1}{3}n^3 - \frac{1}{2}n^2 + \frac{1}{6}n$
$-$	$\frac{1}{3}n^3 - \frac{1}{2}n^2 + \frac{1}{6}n$
$:$	$\frac{1}{2}n^2 - \frac{1}{2}n$

Iterační řešiče používají v prvním kroku počáteční odhad řešení a každá k – tá iterace vychází z minulého spočteného řešení. Tento proces se opakuje až do doby, kdy metoda zkonverguje. Zastavení iterační metody je zajištěno takzvanými zastavovacími kritérii, případně maximálním možným počtem iterací.

V problémech numerické lineární algebry často vznikají matice, které jsou velké a řídké, a mohou být řádu několika miliónů či dokonce miliard. V jiných případech nemusí být matice příliš velká, ale může mít komplikovanou strukturu, může např. obsahovat bloky vyjádřené maticovou funkcí, jež je velmi drahé sestavit. Často pak jedinou operací, kterou jsme schopni s maticí levně a efektivně provést, je násobení matice s vektorem.

Iterační metody si vysvětlíme na řešení klasických systémů lineárních rovnic, ve tvaru $Ax = b$, kde $A \in \mathbb{C}^{n \times n}$ je regulární matice a $b \in \mathbb{C}^n$. Cílem iteračních metod je nalezení dobré aproximace x_k přesného řešení x v co nejmenším počtu iterací. U klasických iteračních metod posuzujeme jejich konvergenci v asymptotickém smyslu (rychlost konvergence je kvantifikována limitou pro počet iterací rostoucí nade všechny meze, přičemž konvergence metody nebere v úvahu možný počáteční přechodový jev). Naproti tomu metody Krylovových podprostorů dávají v přesné aritmetice přesné řešení soustavy v nejvýše n krocích. V praxi nás zajímá dosažení dostatečně přesné aproximace řešení v co nejmenším počtu kroků i a přechodový jev nemá u metod Krylovových podprostorů smysl [10, s. 168].

Mezi klasické iterační metody patří Jacobiho metoda a Gauss-Seidelova metoda. Tyto metody jsou velmi jednoduché, avšak konvergují poměrně pomalu a lze je použít pouze na ostře diagonálně dominantní matice. Mezi často používané iterační metody patří předpomíněné Krylovovy řešiče. Předpokládání soustavy je proces, kdy se snažíme modifikovat původní soustavu rovnic tak, aby metoda použitá na transformovanou úlohu dosáhla mnohem rychlejší konvergence, než při použití na úlohu původní [10]. Mezi oblíbené předpomíňovače patří neúplná LU (ILU) faktori-zace, nebo algebraický multigrid. Mezi Krylovovy řešiče například patří:

CG - metoda sdružených gradientů

BiCGStab - metoda bikonjugovaných gradientů

GMRES - zobecněná minimální reziduální metoda

Metoda sdružených gradientů (CG) řeší soustavy lineárních algebraických rovnic $A\mathbf{x} = \mathbf{b}$ s reálnou symetrickou pozitivně definitní maticí případně s hermitovskou pozitivně definitní maticí (zkratka CG pochází z anglického conjugate gradients). Jde o teoreticky velmi silný nástroj s paměťově nenáročným algoritmem [10, s. 81].

Zobecněná metoda minimálních reziduí (GMRES - z anglického generalized minimal residual method) byla vytvořena pro matice které nejsou pozitivně definitní. Metoda je matematicky ekvivalentní ke generalizované metodě sdružených gradientů a k metodě ORTHODIR.

Metodu bikonjugovaných gradientů lze použít na obecné nehermitovské matice. Její princip je podobný jako u CG, ale není zaručená konvergence.

2.2 Iterační (sekvenčně implicitní) řešení

Iterační řešení je postup, kde je problém proudění a mechaniky rozdělen. V prvním kroku se vyřeší buď problém mechaniky nebo proudění, následně je vyřešen druhý problém již s použitím nejnovějších informací o řešení. V každém časovém kroku se tento postup opakuje až do doby, kdy řešení dosáhne požadované přesnosti.

Nejznámějšími iteračními metodami používanými na soustavy rovnic popisující propojení proudění s mechanikou v porézních médiích jsou: undrained split, fixed-stress split, drained split a fixed stress strain split. Bylo prokázáno, že poslední dvě zmíněné metody mají problémy se stabilitou, narozdíl od prvních dvou metod, které jsou robustní a tedy stabilní.

Metoda undrained split spočívá v uložení konstantní hmotnosti kapaliny a následného počítání tlaků $p^{n+1/2}$ v polovičním časovém kroku a poté p^{n+1} a stanovení $p^{n+1/2} = p^n - \alpha \frac{1}{S} \nabla \cdot (u^{n+1/2} - u^n)$.

Naopak v metodě fixed-stress splitting uložíme konstantní celkové objemové napětí [11]. Metodu fixed-stress split si popíšeme pomocí pseudokódu:

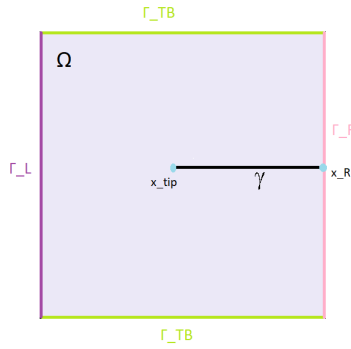
1. Dáno: $(p_h^{n-1}, \mathbf{u}_h^{n-1}), eps, max_{it}$
2. $(p_h^{n,0}, \mathbf{u}_h^{n,0}) := (p_h^{n-1}, \mathbf{u}_h^{n-1})$
3. $Proi = 1, 2, \dots, max_{it}$:
 - Pomocí $(p_h^{n,i-1}, \mathbf{u}_h^{n,i-1})$ spočti $p_h^{n,i}$
 - Pomocí $(p_h^{n,i}, \mathbf{u}_h^{n,i-1})$ spočti $\mathbf{u}_h^{n,i}$
 - Pokud $|p_h^{n,i} - p_h^{n,i-1}|/|p_h^{n,i-1}| < eps$ a $|\mathbf{u}_h^{n,i} - \mathbf{u}_h^{n,i-1}|/|\mathbf{u}_h^{n,i-1}| < eps$, pak jdi na 4.
4. Pokud $i < max_{it}$, pak $(p_h^n, \mathbf{u}_h^n) := (p_h^{n,i}, \mathbf{u}_h^{n,i})$

kde eps je daná přesnost a max_{it} je maximální počet iterací, při kterém se zastaví řešení.

Mezi iteračními schématy byla tato široce používaná metoda prokázána jako bezpodmínečně stabilní ve smyslu Von Neumannovy analýzy a globálně konvergentní, když uvažujeme mírně stlačitelný tok v homogenním porézním médiu [8].

3 Numerické řešení Biotova modelu pro oblast s puklinou

V této kapitole se budeme věnovat řešení Biotova modelu pro oblast s puklinou. Řešíme problém vstřikování tekutiny do pukliny γ ve čtvercovém modelu horniny Ω :



Obrázek 3.1: Model domény s puklinou.

kde $\gamma := (\frac{1}{2}, 1) \times \{\frac{1}{2}\}$ a $\Omega := (0, 1) \times (0, 1)$. Následující model a jeho diskretizace byly odvozeny v diplomové práci Světlany Šaušové [2].

Biotova poroelasticita je popsána systémem rovnic:

$$\left. \begin{aligned} \partial_t(Sp + \alpha \nabla \cdot \mathbf{u}) - \kappa \Delta p &= g \\ -\nabla \cdot (2\mu \epsilon[\mathbf{u}] + \lambda(\nabla \cdot \mathbf{u})\mathbb{I}) + \alpha \nabla p &= \mathbf{f} \end{aligned} \right\} \text{ na } (0, T) \times \Omega$$

kde neznámé jsou tlak p a posunutí \mathbf{u} v oblasti Ω . $\epsilon[\mathbf{u}] := (\nabla \mathbf{u} + \nabla \mathbf{u}^T/2)$ je symetrická část gradientu \mathbf{u} . Storativita S , Biotův koeficient α , hydraulická vodivost κ , Lamého parametry μ , λ , volumetrický zdroj tekutiny g a síla \mathbf{f} jsou dány jako konstantní parametry. Puklina může být prázdná ($\mu = \lambda = 0$), nebo vyplněná elastickým materiálem ($\mu, \lambda > 0$).

Pro redukovaný Biotův model předpokládáme, že tloušťka pukliny δ je vzhledem k velikosti domény zanedbatelná. Po integraci Biotova systému přes šířku pukliny

získáme následující rovnice viz [2]:

$$\begin{aligned} \delta\{\partial_t(S_f p_f + \alpha_f \nabla_\tau \cdot \mathbf{u}_f) - \nabla_\tau \cdot (\kappa_f \nabla_\tau p_f)\} \\ + F^+ + F^- - \partial_t G = \delta g_f \text{ na } (0, T) \times \gamma, \end{aligned} \quad (3.1)$$

$$\begin{aligned} \delta - \nabla_\tau \cdot (2\mu_f \epsilon_\tau[\mathbf{u}_f] + \lambda_f (\nabla_\tau \cdot \mathbf{u}_f) \mathbb{I}) + \alpha_f \nabla_\tau p_f \\ + \mathbf{Q}^+ + \mathbf{Q}^- - \nabla_\tau \cdot \mathbb{R} = \delta \mathbf{f}_f \text{ na } (0, T) \times \gamma. \end{aligned} \quad (3.2)$$

Zde p_f , \mathbf{u}_f jsou neznámé v puklině, S_f , α_f , κ_f , g_f , μ_f , λ_f , \mathbf{f}_f jsou fyzikálními konstantami v puklině, ∇_τ a ϵ_τ značí tečný gradient v γ a jeho symetrickou část. V další části budeme používat toto označení: symbol \mathbf{n}^\pm je jednotkový normálový vektor k γ v horním (+) / dolním (-) směru a p^\pm, \mathbf{u}^\pm značí tlak a posunutí ze spodního p^+ , \mathbf{u}^+ / dolního p^- , \mathbf{u}^- směru. Neznámé F^\pm , \mathbf{Q}^\pm reprezentují toky a normálová napětí působící na obou stranách pukliny a G , \mathbb{R} jsou podmínky vznikající z dimenzionální redukce:

$$F^\pm := F^\pm(p, p_f) = \frac{2}{\delta} \kappa_f (p_f - p^\pm), \quad (3.3)$$

$$G := G(\mathbf{u}) = \alpha_f (\mathbf{u}^- - \mathbf{u}^+) \cdot \mathbf{n}^+, \quad (3.4)$$

$$\begin{aligned} \mathbf{Q}^\pm := \mathbf{Q}^\pm(p, \mathbf{u}, \mathbf{u}_f) = \frac{2}{\delta} (\mu_f (\mathbf{u}_f - \mathbf{u}^\pm) + (\mu_f + \lambda_f) ((\mathbf{u}_f - \mathbf{u}^\pm) \cdot \mathbf{n}^\pm) \mathbf{n}^\pm) \\ + \mu_f \nabla_\tau (\mathbf{u}^\pm \cdot \mathbf{n}^\pm) + \lambda_f (\nabla_\tau \mathbf{u}^\pm) \mathbf{n}^\pm - \alpha_f p^\pm \mathbf{n}^\pm, \end{aligned} \quad (3.5)$$

$$\mathbb{R} := \mathbb{R}(\mathbf{u}) = -\mu_f (\mathbf{u}^- - \mathbf{u}^+) \otimes \mathbf{n}^+ + \lambda_f (\mathbf{u}^- - \mathbf{u}^+) \cdot \mathbf{n}^+ \mathbb{I}. \quad (3.6)$$

Systém (3.1) - (3.2) je spojen s rovnicemi (4) na Ω pomocí podmínek na rozhraní:

$$\begin{aligned} \kappa \nabla p^\pm \cdot \mathbf{n}^\pm = F^\pm, \quad (2\mu \epsilon[\mathbf{u}^\pm] + \lambda (\nabla \cdot \mathbf{u}^\pm) \mathbb{I}) \mathbf{n}^\pm - \alpha p^\pm \mathbf{n}^\pm = \mathbf{Q}^\pm \text{ na } (0, T) \times \gamma. \end{aligned} \quad (3.7)$$

Na špičce pukliny $\mathbf{x}_{tip} := [\frac{1}{2}, \frac{1}{2}]^T$ uvažujeme homogenní Neumannovy podmínky jak pro proudění tak pro mechaniku.

Označme si následující části hranice: $\Gamma_L := \{0\} \times (0, 1)$ jako levou stranu, $\Gamma_{TB} :=$

$(0, 1) \times \{0, 1\}$ jako horní a dolní stranu, $\Gamma_R := \{1\} \times (0, 1)$ jako pravou stranu hranice Ω . Dále nechť $\gamma_R := \{\mathbf{x}_R\}$, kde $\mathbf{x}_R := [1, \frac{1}{2}]^T$ je pravý konec pukliny.

Rovnice Biotova systému na Ω a na γ jsou doplněny o tyto okrajové podmínky:

- Tekutina je vstřikována do pukliny pod daným tlakem p_{in} :

$$p_f(t, \mathbf{x}_R) = p_{in}(t) := \begin{cases} \sin(\frac{\pi}{T}t), & t < T/2, \\ 1, & t \geq T/2 \end{cases} \text{ pro } t \in (0, T).$$

- Tlak na levé straně je pevně daný

$$p = 0 \text{ na } (0, T) \times \Gamma_L. \quad (3.8)$$

- Deformace na horní a dolní straně je rovna nule:

$$\mathbf{u} = \mathbf{0} \text{ na } (0, T) \times \Gamma_{TB}. \quad (3.9)$$

- Pravá strana stejně jako vnější okraj pukliny se může deformovat pouze ve vertikálním směru:

$$u_1 = 0 \text{ na } (0, T) \times \Gamma_R, \quad u_{f1} = 0 \text{ na } (0, T) \times \gamma_R. \quad (3.10)$$

- Ve zbytku hranice uvažujeme homogenní Neumannovy podmínky.

V počátečním čase $t = 0$ je systém v klidu, tedy:

$$p(0, \cdot) = p_f(0, \cdot) = 0 \quad \mathbf{u}(0, \cdot) = \mathbf{u}_f(0, \cdot) = \mathbf{0}. \quad (3.11)$$

Pomocí slabé formulace najdeme řešení $(p, p_f, \mathbf{u}, \mathbf{u}_f)$ v odpovídajících prostorech funkcí uspokojující počáteční a okrajové podmínky takové, že:

$$\frac{d}{dt}(S_p + \alpha \nabla \mathbf{u}, q)_\Omega + \kappa(\nabla p, \nabla q)_\Omega - (F^+, q^+)_\gamma - (F^-, q^-)_\gamma = (g, q)_\Omega, \quad (3.12)$$

$$\begin{aligned} \delta \left\{ \frac{d}{dt}(S_f p_f + \alpha_f \nabla_\tau \cdot \mathbf{u}_f, q_f)_\gamma + \kappa_f(\nabla_\tau p_f, \nabla_\tau q_f)_\gamma \right\} \\ + (F^+ + F^-, q_f)_\gamma - \frac{d}{dt}(G, q_f)_\gamma = \delta(g_f, q_f)_\gamma, \end{aligned} \quad (3.13)$$

$$2\mu(\epsilon[\mathbf{u}], \epsilon[\mathbf{v}])_\Omega + \lambda(\nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v})_\Omega - \alpha(p, \nabla \cdot \mathbf{v})_\Omega - (\mathbf{Q}^+, \mathbf{v}^+)_\gamma - (\mathbf{Q}^-, \mathbf{v}^-)_\gamma = (\mathbf{f}, \mathbf{v})_\Omega, \quad (3.14)$$

$$\delta \left\{ 2\mu_f(\epsilon_\tau[\mathbf{u}_f], \epsilon_\tau[\mathbf{v}_f])_\gamma + \lambda_f(\nabla_\tau \cdot \mathbf{u}_f, \nabla_\tau \cdot \mathbf{v}_f)_\gamma - \alpha_f(p_f, \nabla_\tau \cdot \mathbf{v}_f) \right\} + (\mathbf{Q}^+ + \mathbf{Q}^-, \mathbf{v}_f)_\gamma + (\mathbb{R}, \nabla_\tau \mathbf{v}_f)_\gamma = \delta(\mathbf{f}_f, \mathbf{v}_f)_\gamma \quad (3.15)$$

pro všechny testovací funkce $(q, q_f, \mathbf{v}, \mathbf{v}_f)$ a všechny časy v $(0, T)$. Výraz $(f, g)_\Omega$ zde značí integrál součinu fg přes Ω . Detaily odvození slabé formulace a definice prostorů lze nalézt v [2].

Pro numerické řešení použijeme nespojité Galerkinovy metody pro diskretizaci rovnic na Ω a metodu konečných prvků pro rovnice na γ . Nechť \mathcal{T}_h je triangulace sítě Ω , tj množina vzájemně disjunktních trojúhelníkových prvků tak, že $\cup_{E \in \mathcal{T}_h} \bar{E} = \bar{\Omega}$. Množina všech stran elementů v \mathcal{T}_h bude označena jako \mathcal{E}_h a množina vnitřních stran jako $\mathcal{E}_{h,int}$. Předpokládáme, že \mathcal{T}_h je kompatibilní s γ , tj je zde podmnožina vnitřních stran $\mathcal{E}_{h,\gamma} \subset \mathcal{E}_{h,int}$ tak že $\cup_{F \in \mathcal{E}_{h,\gamma}} \bar{F} = \bar{\gamma}$.

Pro prostorovou diskretizaci na Ω uvažujeme funkce které jsou po částech lineární na elementech z \mathcal{T}_h , ale obecně nespojité na $\bar{\Omega}$. V puklině uvažujeme funkce po částech lineární na okrajích $\mathcal{E}_{h,\gamma}$ a spojité na $\bar{\gamma}$. Důvodem užití nespojitých polí na Ω je, že software založené na metodě konečných prvků jako například FEniCS nepodporují vnitřní nespojitosti podél puklin. Spojitost na rozhraní vnitřních elementů je vynucena dodatečnými podmínky ve formulaci.

Pro dočasnou diskretizaci bereme diskrétní časový krok $\Delta t := T/N$, kde N je pozitivní celé číslo. V diskrétních časech $n\Delta t$ budou funkce označeny horním indexem $()^n$. Časové derivace budou aproximovány implicitní Eulerovou metodou, tj $\partial_t f|_{t=n\Delta t} \approx \frac{f^n - f^{n-1}}{\Delta t}$.

Diskrétní problém je: Najděte $\{(p_h^n, p_{fh}^n, \mathbf{u}_h^n, \mathbf{u}_{fh}^n)\}_{n=1}^N$ tak, že

$$\begin{aligned} & \frac{1}{\Delta t} (Sp_h^n + \alpha \nabla \cdot \mathbf{u}_h^n, q_h)_\Omega + \kappa (\nabla p_h^n, \nabla q_h)_\Omega - (F_h^{n+}, q_h^+)_\gamma - (F_h^{n-}, q_h^-)_\gamma \\ & - \sum_{F \in \mathcal{E}_{h,int}} \left((\kappa \{\{\nabla p_h^n\}\}, \llbracket q_h \rrbracket \mathbf{n}^+)_F - (\kappa \llbracket p_h^n \rrbracket, \{\{q_h\}\} \mathbf{n}^+)_F \right) \\ & + \sum_{F \in \mathcal{E}_{h,int}} \left(\Gamma \frac{\kappa}{|F|} \llbracket p_h^n \rrbracket, \llbracket q_h \rrbracket \right)_F = (g, q_h)_\Omega + \frac{1}{\Delta t} (Sp_h^{n-1} + \alpha \nabla \cdot \mathbf{u}_h^{n-1}, q_h)_\Omega, \end{aligned} \quad (3.16)$$

$$\begin{aligned}
& \delta \left\{ \frac{1}{\Delta t} (S_f p_{fh}^n + \alpha_f \nabla_\tau \cdot \mathbf{u}_{fh_\tau}^n, q_{fh})_\gamma + \kappa_f (\nabla_\tau p_{fh}^n, \nabla_\tau q_{fh})_\gamma \right\} \\
& \quad + (F_h^{n+} + F_h^{n-}, q_f)_\gamma - \frac{1}{\Delta t} (G_h^n, q_f)_\gamma \\
& = \gamma \left\{ (g_f, q_f)_\gamma + \frac{1}{\Delta t} (S_f p_{fh}^{n-1} + \alpha_f \nabla_\tau \mathbf{u}_{fh}^{n-1}, q_{fh})_\gamma \right\} - \frac{1}{\Delta t} (G_h^{n-1}, q_f)_\gamma,
\end{aligned} \tag{3.17}$$

$$\begin{aligned}
& 2\mu(\epsilon[\mathbf{u}_h^n], \epsilon[\mathbf{v}_h])_\Omega + \lambda(\nabla \cdot \mathbf{u}_h^n, \nabla \cdot \mathbf{v}_h)_\Omega - \alpha(p_h^n, \nabla \cdot \mathbf{v}_h)_\Omega \\
& \quad - (\mathbf{Q}_h^{n+}, \mathbf{v}_h^+)_\gamma - (\mathbf{Q}_h^{n-}, \mathbf{v}_h^-)_\gamma \\
& - \sum_{F \in \mathcal{E}_{h,int}} \left((\lambda \{ \{ \nabla \cdot \mathbf{u}_h^n \} \}, \llbracket \mathbf{v}_h \rrbracket \mathbf{n}^+)_F - (\lambda \llbracket \mathbf{u}_h^n \rrbracket, \{ \{ \nabla \cdot \mathbf{v}_h \} \} \mathbf{n}^+)_F \right. \\
& \quad \left. + 2\mu(\{ \{ \epsilon[\mathbf{u}_h^n] \} \} \mathbf{n}^+, \llbracket \mathbf{v}_h \rrbracket)_F - 2\mu(\llbracket \mathbf{u}_h \rrbracket, \{ \{ \epsilon[\mathbf{v}_h] \} \} \mathbf{n}^+)_F \right) \\
& \quad + \sum_{F \in \mathcal{E}_{h,int}} (\Gamma \frac{\mu + \lambda}{|F|} \llbracket \mathbf{u}_h^n \rrbracket, \llbracket \mathbf{v}_h \rrbracket)_F = (\mathbf{f}, \mathbf{v}_h)_\Omega,
\end{aligned} \tag{3.18}$$

$$\begin{aligned}
& \delta \{ 2\mu_f \epsilon_\tau[\mathbf{u}_{fh}^n], \epsilon_\tau[\mathbf{v}_{fh}] \}_\gamma + \lambda_f (\nabla_\tau \cdot \mathbf{u}_{fh}^n, \nabla_\tau \cdot \mathbf{v}_{fh})_\gamma - \alpha_f (p_{fh}^n, \nabla_\tau \mathbf{v}_{fh})_\gamma \} \\
& \quad + (\mathbf{Q}_h^{n+} + \mathbf{Q}_h^{n-}, \mathbf{v}_{fh})_\gamma + (\mathbb{R}_h^n, \nabla_\tau \mathbf{v}_{fh})_\gamma = \delta(\mathbf{f}_f, \mathbf{v}_{fh})_\gamma,
\end{aligned} \tag{3.19}$$

pro všechny testovací funkce $(q_h, q_{fh}, \mathbf{v}_h, \mathbf{v}_{fh})$ a $n = 1, 2, \dots, N$. Zde $F_h^{n\pm}, G_h^n, \mathbf{Q}_h^{n\pm}$ a \mathbb{R}_h^n znamená $F^\pm(p_h^n, p_{fh}^n), G(\mathbf{u}_h^n), \mathbf{Q}^\pm(p_h^n, \mathbf{u}_h^n, \mathbf{u}_{fh}^n)$ a $\mathbb{R}(\mathbf{u}_h^n)$. Dále $\{\{f\}\} := (f^+ + f^-)/2$, $\llbracket f \rrbracket = f^+ - f^-$, a $\Gamma > 0$ je uživatelem definovaný parametr (např. $\Gamma = 10^3$).

Způsob řešení byl implementován do již existujícího výpočetního kódu pro redukovaný Biotův systém. Abstraktní zápis problému (3.16)-(3.19) vypadá v kódu takto:

$$a(\mathbf{s}, \mathbf{t}) = l(\mathbf{t}) \tag{3.20}$$

kde \mathbf{s} je vektor řešení a \mathbf{t} vektor testovacích funkcí.

Numerické řešení Biotova modelu je implementováno pomocí knihovny FEniCS - programování konečných prvků neboli Finite Element (FE) Programming v jazyce Python. FEniCS je open-source výpočetní platforma založená na řešení parciálních diferenciálních rovnic (Partial Differential Equations - PDE), která umožňuje uživatelům přeložit matematické modely do efektivního kódu konečných prvků (FE). Ve výpočetní platformě FEniCS je možné programovat v C++ i v Pythonu, ale pro náš případ

byl použit jazyk Python.

Všechny výpočty byly provedeny na počítači s procesorem Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80GHz a operační pamětí 8GB.

Při výpočtech byly používány tyto hodnoty fyzikálních parametrů:

Tabulka 3.1: Použité parametry v jednotkách SI.

Parametr	Hodnota
T	0,01
δ	0,1
k	0,1
k_f	1
S	0,01
S_f	0,1
f	(0;0)
f_f	(0;0)
μ	1
μ_f	0,5
λ	1
λ_f	0,5
α	1
α_f	1

3.1 Monolitické řešení

Jak již bylo vysvětleno v kapitole 2.1, monolitické řešení soustavy znamená řešení soustavy v celku. Rovnice jsou zde zapsány ve formách a, l . Ve formě a je součet levé strany rovnic (3.16) - (3.17), ve formě l je součet pravé strany rovnic (3.16) a (3.17).

Ve výpočetní knihovně FEniCS je výpočet tedy zapsán v této formě:

```
start_solve = time.time()
solve(a == l, w, bcs, solver_parameters={'linear_solver': bicgstab, 'preconditioner': ilu})
end_solve = time.time()
w0.vector()[:] = w.vector()
```

Obrázek 3.2: Monolitické řešení zapsané v knihovně FEniCS

kde solver parameters určuje parametry našeho řešiče: nejdříve vybíráme samostatný řešič, poté předpomiňovač.

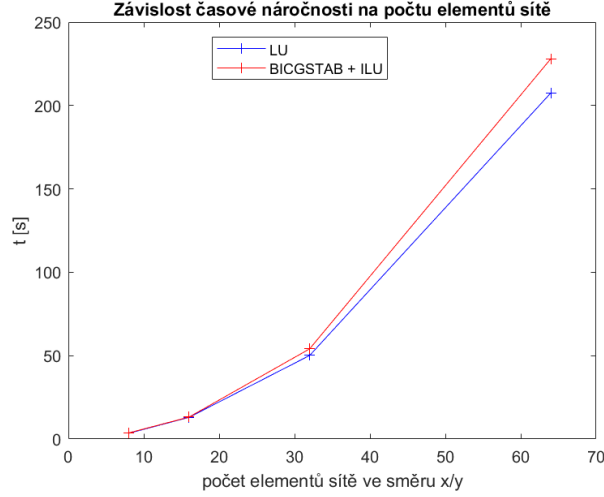
Pro náš problém jsme srovnávali časovou náročnost LU faktorizace (bez předpodmiňovače) a metody bikonjugovaných gradientů (předpodmiňovač ILU - nekompletní LU faktorizace). Další zkoušená metoda, GMRES, na těchto rovnicích nezkonvergovala. Měříme čas pouze samotného výpočtu, obě metody byly spuštěny na intervalu deseti časových kroků. Z těchto deseti kroků byl poté vypočítán průměrný čas výpočtu a směrodatná odchylka. Metody jsme srovnávali pro různé velikosti sítě, kde n_x je počet elementů sítě ve směru x/y.

Tabulka 3.2: Srovnání časového řešení řešičů s předpodmiňovači v závislosti na velikosti sítě

	Řešič, předpodmiňovač		Řešič, předpodmiňovač	
	LU, žádný		BICGSTAB, ILU	
nx	t_průměr [s]	směrodatná odchylka	t_průměr [s]	směrodatná odchylka
8	3,44	0,033	3,67	0,183
16	12,99	0,118	13,16	0,348
32	50,11	3,041	54,07	0,541
64	207,57	2,638	228,08	1,357

V tabulce 3.1 je vidět, že z tohoto srovnání je rychlejší LU metoda. BiCGstab je na tomto modelu pomalejší, vzhledem k tomu, že ILU je sice robustní, ale ne ideální předpodmiňovač. Je potřeba ale myslet na to, že LU faktorizace není použitelná na velké soustavy rovnic.

Maximální velikost sítě byla $64 \times 64 \times 2$ elementů, a časový nárůst řešení v závislosti na nx byl u LU faktorizace a metody bikonjugovaných gradientů kvadratický, můžeme tedy očekávat, že LU faktorizace bude tedy i pro větší síť rychlejším řešičem, dokud nenastane problém zaplnění paměti.



Graf. 3.1: Časové srovnání dvou monolitických metod

3.2 Aplikace fixed stress splitting

Řešení Biotova systému iteračním způsobem je přístup, který zahrnuje postupné implicitní řešení proudění a elasticity s využitím nejnovějších informací o řešení, kdy iterace probíhá v každém časovém kroku až do konvergence.

Namísto řešení systému (3.15) - (3.18) plně propojeným způsobem, je oblíbenou alternativou použití iteračních metod, které oddělují mechaniku od problému s prouděním a umožňují efektivní řešení a předpomínění jednotlivých podproblémů. Zde omezuje naše úvahy na široce používanou metodu fixed stress splitting a přizpůsobíme myšlenku Mikeliće a Wheelerové [11], která považuje zachování umělé objemové konstanty napětí za umělou [8].

Iterační schéma definuje sekvenci $(p^{n,i}, \mathbf{u}^{n,i}, p_f^{n,i}, \mathbf{u}_f^{n,i})$, kde $i \geq 0$. Po inicializaci $\mathbf{u}_h^{n,0} = \mathbf{u}_h^{n-1}, p_h^{n,0} = p_h^{n-1}$ je každý iterát definován ve dvou krocích. Prvně je problém proudění řešen nezávisle. Poté je řešen problém mechaniky (posunutí) pomocí nově spočítaného tlaku v doméně a v puklině. Do rovnic byl přidán relaxační parametr β . Formy a a l lze rozdělit podle rovnic:

$$a = a_{flow} + a_{mech} \quad (3.21)$$

$$l = l_{flow} + l_{mech} \quad (3.22)$$

Výpočet tlaku a posunutí v abstraktním zápisu:

$$\begin{aligned} a_{flow}(\mathbf{s}_{flow}, \mathbf{t}_{flow}) &= l_{flow}(\mathbf{t}_{flow}), \\ a_{mech}(\mathbf{s}_{mech}, \mathbf{t}_{mech}) &= l_{mech}(\mathbf{t}_{mech}), \end{aligned} \quad (3.23)$$

kde $\mathbf{s}_{flow} := (p_h^{n,i}, p_{fh}^{n,i})$ a $\mathbf{s}_{mech} := (\mathbf{u}_h^{n,i}, \mathbf{u}_{fh}^{n,i})$ Pro fixní $n, i \in \mathbb{N}$ vypadá detailní schéma:

$$\begin{aligned} a_{flow} &= \frac{1}{\Delta t}((S + \beta)p_h^n, q_h)_\Omega + \kappa(\nabla p_h^n, \nabla q_h)_\Omega \\ &+ \delta \left\{ \frac{1}{\Delta t}((S_f + \beta_f)p_{fh}^n, q_{fh})_\gamma + \kappa_f(\nabla_\tau p_{fh}^n, \nabla_\tau q_{fh})_\gamma \right\} \\ &- (F_h^{n+}, q_h^+)_\gamma - (F_h^{n-}, q_h^-)_\gamma \\ &- \sum_{F \in \epsilon_{h,int}} \left((\kappa\{\{\nabla p_h^n\}\}, \llbracket q_h \rrbracket \mathbf{n}^+)_F - (\kappa\llbracket p_h^n \rrbracket, \{\{q_h\}\} \mathbf{n}^+)_F \right) \\ &+ \sum_{F \in \epsilon_{h,int}} \left(\Gamma \frac{\kappa}{|F|} \llbracket p_h^n \rrbracket, \llbracket q_h \rrbracket \right)_F, \end{aligned} \quad (3.24)$$

$$\begin{aligned} l_{flow} &= (g, q_h)_\Omega + \frac{1}{\Delta t}(Sp_h^{n-1} + \alpha \nabla \cdot \mathbf{u}_h^{n-1}, q_h)_\Omega \\ &+ \delta \left\{ (g_f, q_f)_\gamma + \frac{1}{\Delta t}(S_f p_{fh}^{n-1} + \alpha_f \nabla_\tau \cdot \mathbf{u}_{fh}^{n-1}, q_{fh})_\gamma \right\} \\ &- \frac{1}{\Delta t}(G_h^{n-1}, q_{fh})_\gamma \\ &- \frac{1}{\Delta t}(\alpha \nabla \cdot \mathbf{u}_h^{n,i-1}, q_h) \\ &- \delta \left(\frac{1}{\Delta t} \alpha_f \nabla_\tau \cdot (\mathbf{u}_f^{n,i-1}, q_f)_\gamma \right) \\ &+ \frac{1}{\Delta t}(G_h^{n,i-1}, q_{fh})_\gamma \\ &+ \frac{1}{\Delta t}(\beta p_h^{n,i-1}, q_h)_\Omega \\ &+ \frac{\delta}{\Delta t}(\beta_f p_{fh}^{n,i-1}, q_{fh})_\gamma, \end{aligned} \quad (3.25)$$

$$\begin{aligned}
\mathbf{q}_{mech_n}^{n^\pm} &= \frac{2}{\delta} (\mu_f (\mathbf{u}_{fh}^n - \mathbf{u}_h^{n^\pm}) + (\mu_f + \lambda_f) ((\mathbf{u}_{fh}^n - \mathbf{u}_h^{n^\pm}) \cdot \mathbf{n}^\pm) \mathbf{n}^\pm) \\
&\quad + \mu_f \nabla_\tau \cdot (\mathbf{u}_h^\pm \cdot \mathbf{n}^\pm) \mathbf{n}^\pm \\
&\quad + \lambda_f \nabla_\tau \cdot (\mathbf{u}_h^{n^\pm} \cdot \mathbf{n}^\pm) \mathbf{n}^\pm
\end{aligned} \tag{3.26}$$

$$\begin{aligned}
a_{mech} &= 2\mu(\epsilon[\mathbf{u}_h^n], \epsilon[\mathbf{v}_h])_\Omega + \lambda(\nabla \cdot \mathbf{u}_h^n, \nabla \cdot \mathbf{v}_h)_\Omega \\
&\quad \delta \left\{ 2\mu_f(\epsilon_\tau[\mathbf{u}_{fh}^n], \epsilon_\tau[\mathbf{v}_{fh}])_\gamma + \lambda_f(\nabla_\tau \cdot \mathbf{u}_{fh}^n, \nabla_\tau \cdot \mathbf{v}_{fh})_\gamma \right\} \\
&\quad + (\mathbb{R}_h^n, \nabla_\tau \mathbf{v}_{fh})_\gamma \\
&\quad + (\mathbf{q}_{mechh}^{n+}, \mathbf{v}_h^+)_\gamma + (\mathbf{q}_{mechh}^{n-}, \mathbf{v}_h^-)_\gamma \\
&\quad - \sum_{F \in \epsilon_{h,int}} (\lambda \{ \{ \nabla \cdot \mathbf{u}_h^n \} \}, \llbracket \mathbf{v}_h \rrbracket \mathbf{n}^+)_F - (\lambda \llbracket \mathbf{u}_h^n \rrbracket, \{ \{ \nabla \cdot \mathbf{v}_h \} \} \mathbf{n}^+)_F \\
&\quad + 2\mu(\{ \{ \epsilon[\mathbf{u}_h^n] \} \} \mathbf{n}^+, \llbracket \mathbf{v}_h \rrbracket)_F - 2\mu(\llbracket \mathbf{u}_h^n \rrbracket, \{ \{ \epsilon[\mathbf{v}_h] \} \} \mathbf{n}^+)_F \\
&\quad + \sum_{F \in \epsilon_{h,int}} (\Gamma \frac{\mu + \lambda}{|F|} \llbracket \mathbf{u}_h^n \rrbracket, \llbracket \mathbf{v}_h^n \rrbracket)_F
\end{aligned} \tag{3.27}$$

$$\begin{aligned}
l_{mech} &= (\mathbf{f}, \mathbf{v}_h)_\Omega + \delta(\mathbf{f}_f, \mathbf{v}_{fh})_\gamma \\
&\quad + (\alpha p_h^{n,i-1} \nabla \cdot \mathbf{v}_h)_\Omega \\
&\quad + (\delta \alpha_f p_{fh}^{n,i-1} \nabla_\tau \cdot \mathbf{v}_{fh})_\gamma \\
&\quad + \alpha_f (p_h^{(n,i-1)+} \mathbf{n}^+, \mathbf{v}_{fh} - \mathbf{v}^+) \\
&\quad + \alpha_f (p_h^{(n,i-1)-} \mathbf{n}^-, \mathbf{v}_{fh} - \mathbf{v}^-)
\end{aligned} \tag{3.28}$$

Pro rovnice uvažujeme tři relaxační parametry: fyzikálně motivovaný parametr β_{cl} a parametry $\beta_\lambda, \beta_{opt}$, odvozené analýzou Mikélica a Wheelera [11], které jsou platné pro homogenní Lamého parametry. Parametry jsou dány takto:

$$\beta_{cl} = \frac{\alpha^2}{\frac{2\mu}{d} + \lambda}, \quad \beta_\lambda = \frac{\alpha^2}{2\lambda}, \quad \beta_{opt} = \frac{\alpha^2}{2(\frac{2\mu}{d} + \lambda)}. \tag{3.29}$$

$$\beta_{clf} = \frac{\alpha_f^2}{\frac{2\mu_f}{d_f} + \lambda_f}, \quad \beta_{\lambda f} = \frac{\alpha_f^2}{2\lambda_f}, \quad \beta_{fopt} = \frac{\alpha_f^2}{2(\frac{2\mu_f}{d_f} + \lambda_f)}. \tag{3.30}$$

kde d, d_f je dimenze. Biotův model je 2D model, tedy dimenze domény je $d = 2$, a dimenze pukliny je $d_f = 1$.

```

start_solve = time.time()
norm_flow = 1
norm_mech = 1
ite = 0
while ((norm_flow > eps) or (norm_mech > eps)) and ite < 1000:
    #solve linear system for flow
    solve(a_flow == l_flow, wp, bc_flow, solver_parameters={'linear_solver': 'lu'})
    #compute the difference of i and i-1 flow vectors
    if ite > 1:
        norm_flow = ((wp.vector() - wi0_flow.vector()).norm('l2') / wp.vector().norm('l2'))
    else:
        norm_flow = 1
    #update solution of flow within one time step
    wi0_flow.vector()[:] = wp.vector()
    #solve linear system for mech
    solve(a_mech == l_mech, wu, bc_mech, solver_parameters={'linear_solver': 'lu'})
    #compute the difference of i and i-1 mech vectors
    if ite > 1:
        norm_mech = ((wu.vector() - wi0_mech.vector()).norm('l2') / wu.vector().norm('l2'))
    else:
        norm_mech = 1
    # update solution of mech within one time step
    wi0_mech.vector()[:] = wu.vector()
    ite = ite + 1
end_solve = time.time()
#update solution after each time step
w0_flow.vector()[:] = wi0_flow.vector()
w0_mech.vector()[:] = wi0_mech.vector()
end_solve = time.time()

```

Obrázek 3.3: Iterační řešení zapsané v knihovně FEniCS.

Metodu fixed stress splitting můžeme zapsat pomocí pseudokódu takto:

1. Dáno: $(p_h^{n-1}, p_{fh}^{n-1}, \mathbf{u}_h^{n-1}, \mathbf{u}_{fh}^{n-1}), eps, max_{it}$
2. $(p_h^{n,0}, p_{fh}^{n,0}, \mathbf{u}_h^{n,0}, \mathbf{u}_{fh}^{n,0}) := (p_h^{n-1}, p_{fh}^{n-1}, \mathbf{u}_h^{n-1}, \mathbf{u}_{fh}^{n-1})$
3. $Proi = 1, 2, \dots, max_{it}$:
 - Pomocí $((p_h^{n-1}, p_{fh}^{n-1}, \mathbf{u}_h^{n-1}, \mathbf{u}_{fh}^{n-1})$ spočti $(p_h^{n,i}, p_{fh}^{n,i})$
 - Pomocí $(p_h^{n,i}, p_{fh}^{n,i}, \mathbf{u}_h^{n,i-1}, \mathbf{u}_{fh}^{n,i-1})$ spočti $(\mathbf{u}_h^{n,i}, \mathbf{u}_{fh}^{n,i})$
 - Pokud $|(p_h^{n,i}, p_{fh}^{n,i}) - (p_h^{n,i-1}, p_{fh}^{n,i-1})| / |(p_h^{n,i-1}, p_{fh}^{n,i-1})| < eps$ a $|\mathbf{u}_h^{n,i}, \mathbf{u}_{fh}^{n,i}) - (\mathbf{u}_h^{n,i-1}, \mathbf{u}_{fh}^{n,i-1})| / |(\mathbf{u}_h^{n,i-1}, \mathbf{u}_{fh}^{n,i-1})| < eps$, pak jdi na 4.
4. Pokud $i < max_{it}$, pak $(p_h^n, p_{fh}^n, \mathbf{u}_h^n, \mathbf{u}_{fh}^n) := (p_h^{n,i}, p_{fh}^{n,i}, \mathbf{u}_h^{n,i}, \mathbf{u}_{fh}^{n,i})$

Po aplikaci iteračního schématu jsme ověřovali přesnost řešení. Vzhledem k tomu, že rovnice nemají analytické řešení, chyba se počítá jako absolutní hodnota z rozdílu od monolitického řešení. Pro tlak je výpočet následující:

$$e_p = \frac{|p^{ite} - p^{mono}|}{|p^{mono}|} \quad (3.31)$$

kde p^{ite}, p^{mono} je vektor stupňů volnosti pro tlak v Ω spočtený iteračně, resp. monoliticky, přičemž monolitické řešení se považuje za přesné, a obdobně pro posunutí.

Tabulka 3.3: Absolutní hodnota relativní chyby iteračního schématu oproti monolitickému řešení v konečném čase v závislosti na zvolené přesnosti.

	$ mono - ite $			
eps	e_p	e_{pf}	e_u	e_{uf}
1,00E-05	2,40E-06	8,30E-08	1,00E-06	8,30E-08
1,00E-06	5,60E-07	8,30E-08	3,30E-08	5,00E-09
1,00E-07	4,20E-08	5,50E-09	2,30E-09	2,90E-10
1,00E-08	2,00E-09	2,20E-10	9,80E-11	1,10E-11

Naším cílem bylo mít tak přesné řešení, že chyba bude menší než 10^{-6} v tlaku i posunutí (v doméně i v puklině). Z tabulky 3.2 je vidět, že tento požadavek splňuje $eps = 10^{-6}$. Největší chyba se vždy vyskytovala v doméně, přičemž tlak p byl naší určující hodnotou, podle které jsme přesnost řešení určovali. V puklině byla vždy odchylka menší, než v doméně. Z tabulky 3.2 je vidět, že přesnost řešení roste, a tak můžeme usoudit, že iterační schéma je konvergentní. Výpočty přesnosti byly provedeny v ParaView, pomocí funkce Calculator. ParaView je open-source aplikace pro analýzu a vizualizaci.

Souběžně se zjišťováním postačující přesnosti jsme testovali, který parametr β je pro iterační schéma ten nejvhodnější. Prvním testem bylo srovnání iterací a časové náročnosti všech tří parametrů v závislosti na velikosti sítě.

Tabulka 3.4: Měření počtu iterací a časové náročnosti parametrů v závislosti na velikosti sítě při $eps = 10^{-6}$.

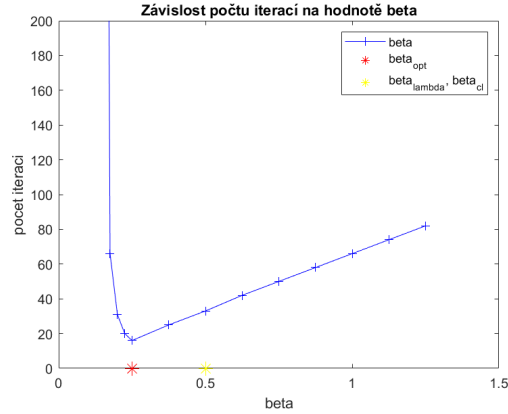
	β_λ	β_{cl}	β_{opt}
nx	n ite	n ite	n ite
16	37	33	16
32	34	33	16
64	34	33	17

Prvním poznatkem je, že jakýkoliv relaxační parametr β zajišťuje stabilní počet iterací, nezávisle na velikosti sítě. Druhým poznatkem je fakt, že při použití β_{opt} klesl počet iterací o polovinu vzhledem k β_{cl} . Vzhledem k výsledkům z tohoto testu jsme usoudili, že pro analýzu konvergence použijeme relaxační parametr β_{opt} .

3.3 Analýza konvergence

V této kapitole se soustředíme na vliv různých parametrů výpočetního modelu na počet iterací.

Naším cílem je zjistit, jaký násobek relaxačního parametru β_{opt} je ten nejvhodnější, proto jsme provedli test závislosti počtu iterací na hodnotách relaxačního parametru $\beta := c \cdot \beta_{opt}$ pro násobek c ve zvoleném rozsahu: $[\frac{1}{2}; \frac{3}{5}; \frac{7}{10}; \frac{4}{5}; 1; \frac{3}{2}; 2; \frac{5}{2}; 3; \frac{7}{2}; 4; \frac{9}{2}; 5]$. Zde mluvíme o násobku c obou parametrů β a β_f , vzhledem k tomu, že parametry vždy měníme zároveň a jejich vliv na počet iterací je propojen.

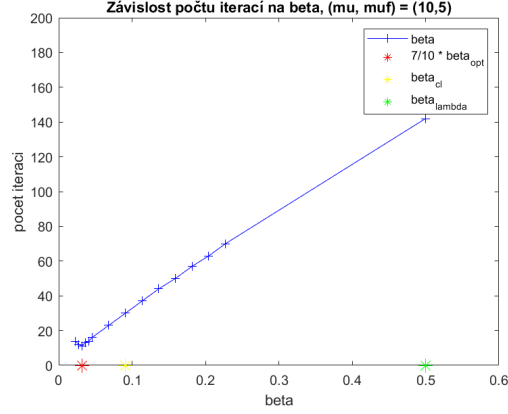


Graf. 3.2: Závislost počtu iterací na hodnotě beta při $nx = 16$ a $eps = 10^{-6}$.

Z grafu 3.2 vidíme, že pokud se relaxační parametr rovná nule, počet iterací se limitně blíží k nekonečnu; takovýto kód obvykle nezkonverguje. Nejmenší počet iterací β je v hodnotě $\beta = 0,25$, což je rovno $1 * \beta_{opt}$, dále hodnoty lineárně stoupají. Vzhledem ke zvoleným parametrům v grafu 3.3 splývají hodnoty $\beta_{cl}, \beta_{\lambda}$.

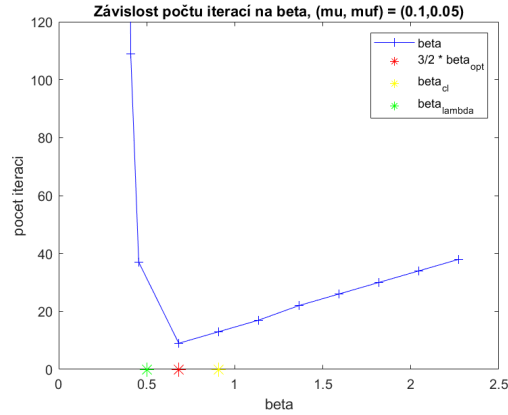
V další analýze jsme zkoumali vliv parametrů Biotova modelu, které mají přímý vliv na parametr β . Relaxační parametr β závisí na těchto třech parametrech: α , μ a λ , a ekvivalentně parametr β_f závisí na α_f , μ_f a λ_f , je-li $\alpha = \alpha_f = 0$, pak rovnice elasticity a rovnice proudění nejsou propojené. Víme, že s rostoucím α, α_f roste i počet iterací. V dalších testech jsme zjišťovali souvislost mezi těmito parametry výpočetního modelu a relaxačním parametrem. Následují testy, kdy jsme postupně měnili hodnoty těchto fyzikálních parametrů: (μ, μ_f) , (λ, λ_f) , (λ_f, μ_f) , $(\lambda, \lambda_f, \mu, \mu_f)$, (κ, κ_f) .

Při závislosti na změně druhého Lamého parametru (μ, μ_f) jsme postupně prověřili dvojice parametrů $(1; 0, 5)$, $(10; 5)$, $(0, 1; 0, 05)$ a $(1; 1)$, $(1; 0, 1)$, $(1; 0, 01)$. První dvojice parametrů $(1; 0, 5)$ je původní sada parametrů, tedy graf závislosti počtu iterací na parametru β je stejný jako graf 3.2.



Graf. 3.3: Závislost počtu iterací na parametru β při druhém Lamého parametru $(\mu, \mu_f) = (10; 5)$.

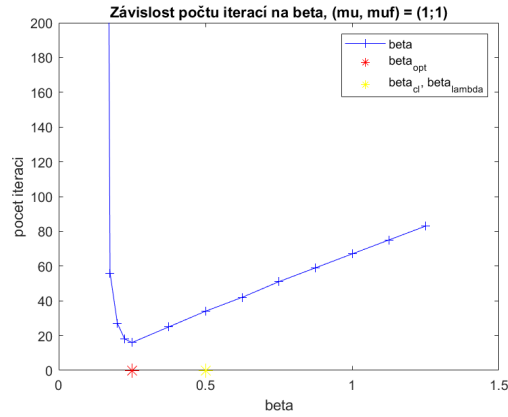
Z grafu 3.3 můžeme vidět, že při změně velikosti druhého Lamého parametru v doméně i v puklině se změní i nejvhodnější hodnota relaxačního parametru β_{opt} , z původního $c = 1$ na $c = \frac{7}{10}$. Pokud se tedy zvětší druhý Lamého parametru, klesne hodnota optimálního násobku relaxačního parametru a počet iterací s velikostí hodnoty β prudce lineárně roste. β_λ i β_{cl} při této sadě druhých Lamého parametru poskytují větší počet iterací.



Graf. 3.4: Závislost počtu iterací na parametru β při druhém Lamého parametru $(\mu, \mu_f) = (0, 1; 0, 05)$.

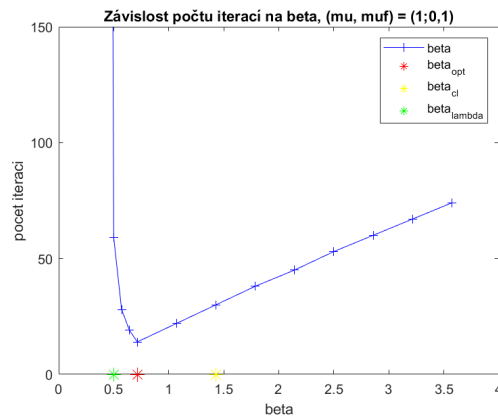
Z grafu 3.4 vidíme, že při $(\mu, \mu_f) = (0, 1; 0, 05)$ je nejvhodnějším násobkem $c = 3/2$. Při zmenšení druhého Lamého parametru se také parametry β_λ a β_{cl} přiblíží k optimální hodnotě relaxačního parametru.

Ve druhém testu byl konstantní parametr μ a postupně zmenšován parametru μ_f tak, aby poměr těchto parametru byl postupně $1, \frac{1}{10}, \frac{1}{100}$.



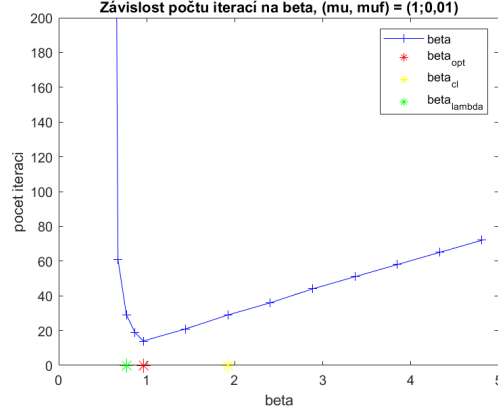
Graf. 3.5: Závislost počtu iterací na parametru β při druhém Lamého parametru $(\mu, \mu_f) = (1; 1)$.

Z grafu 3.5 vidíme, že při $\mu = \mu_f = 1$ zůstává nejvhodnějším násobkem parametru β_{opt} $c = 1$. Vzhledem ke zvoleným parametrům v grafu splývají hodnoty β_{cl} a β_{λ} .



Graf. 3.6: Závislost počtu iterací na parametru β při druhém Lamého parametru $(\mu, \mu_f) = (1; 0, 1)$.

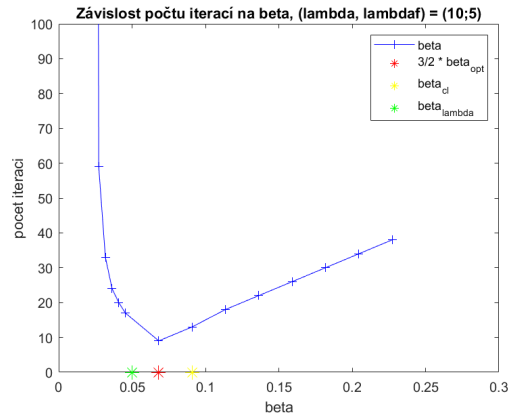
Z grafu 3.6 vidíme, že při $(\mu, \mu_f) = (1; 0, 1)$ zůstává nejvhodnějším násobkem parametru β_{opt} $c = 1$. β_{λ} poskytuje stejný počet iterací jako β_{cl} , i když má každý parametr jinou hodnotu.



Graf. 3.7: Závislost počtu iterací na parametru β při druhém Lamého parametru $(\mu, \mu_f) = (1; 1)$.

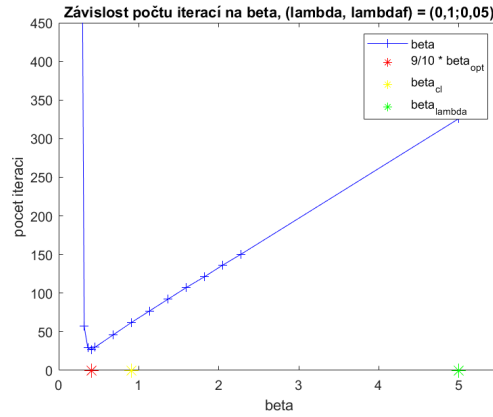
Z grafu 3.7 vidíme, že nejvhodnějším parametrem je stále $1 \cdot \beta_{opt}$. Pokud srovnáme grafy 3.5 - 3.6, vidíme, že při změně poměru druhého Lamého parametru v doméně a v puklině zůstává hodnota nejlepšího násobku a relaxační parametr stabilní.

Dále jsme proměřili závislost relaxačního parametru na změně prvního Lamého parametru (λ, λ_f) . Postupně jsme tedy měnili parametry: $(1; 0, 5)$, $(10; 5)$, $(0, 1; 0, 05)$ a $(1; 1)$, $(1; 0, 1)$, $(1; 0, 01)$. První dvojice parametrů $(1; 0, 5)$ je původní sada parametrů, tedy graf závislosti počtu iterací na parametru β je stejný jako graf 3.2.



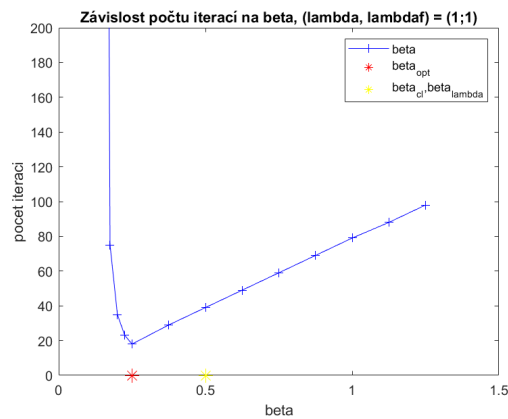
Graf. 3.8: Závislost počtu iterací na parametru β při prvním Lamého parametru $(\lambda, \lambda_f) = (10; 5)$.

Z grafu 3.8 vidíme, že při zvětšení prvního Lamého parametru v doméně i v puklině se také zvětší násobek optimálního relaxačního parametru $c = 3/2$. β_{cl} v tomto případě poskytuje stejný počet iterací jako β_{λ} .



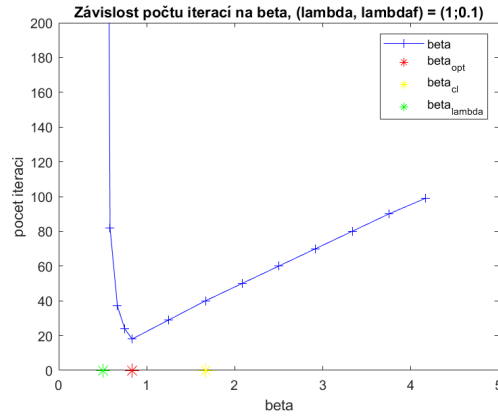
Graf. 3.9: Závislost počtu iterací na parametru β při prvním Lamého parametru $(\lambda, \lambda_f) = (0, 1; 0, 05)$.

Při zmenšení prvního Lamého parametru v doméně i v puklině se i zmenší c na $c = 9/10$, viz graf 3.9.

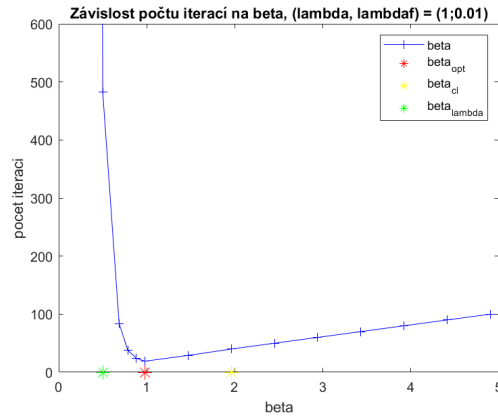


Graf. 3.10: Závislost počtu iterací na parametru β při prvním Lamého parametru $(\lambda, \lambda_f) = (1; 1)$.

Z grafu 3.10 je vidět, že při $\lambda = \lambda_f = 1$ je optimálním násobkem $c = 1$. Vzhledem ke zvoleným parametrům zde splývají hodnoty β_λ a β_{cl} .

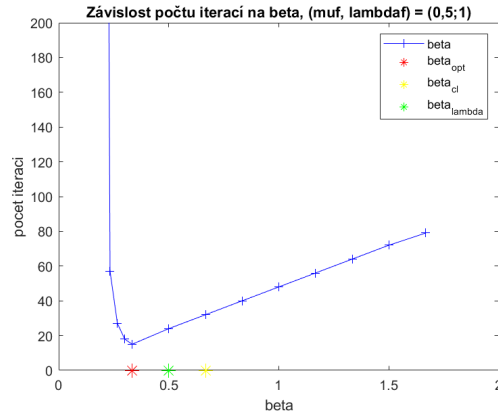


Graf. 3.11: Závislost počtu iterací na parametru β při prvním Lamého parametru $(\lambda, \lambda_f) = (1; 0, 1)$.



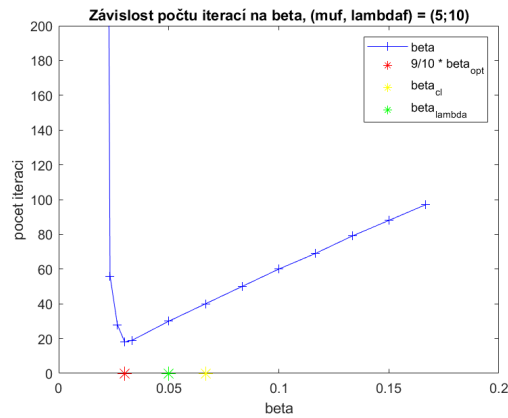
Graf. 3.12: Závislost počtu iterací na parametru β při prvním Lamého parametru $(\lambda, \lambda_f) = (1; 0, 01)$.

Z grafů 3.11 a 3.12 vidíme, že zmenšení velikosti λ_f nemá vliv na změnu c . Násobek optimálního relaxačního parametru zůstává $c = 1$. Dále jsme provedli test závislosti počtu iterací na Lamého parametrech pouze v puklině.



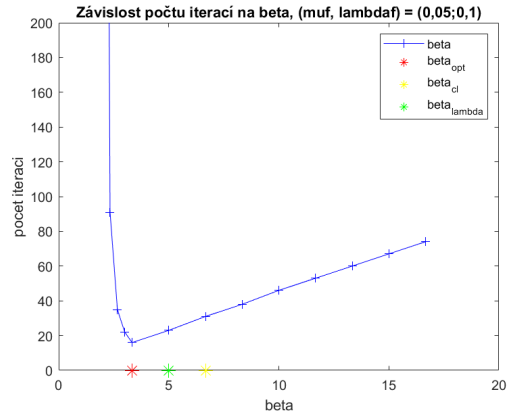
Graf. 3.13: Vliv prvního a druhého Lamého parametru v puklině na počet iterací, $(\lambda_f, \mu_f) = (1; 0,5)$.

Z grafu 3.13 vidíme, že při změně Lamého parametru v puklině je nejlepší násobek parametru β_{opt} roven $c = 1$.



Graf. 3.14: Vliv prvního a druhého Lamého parametru v puklině na počet iterací, $(\lambda_f, \mu_f) = (10; 5)$

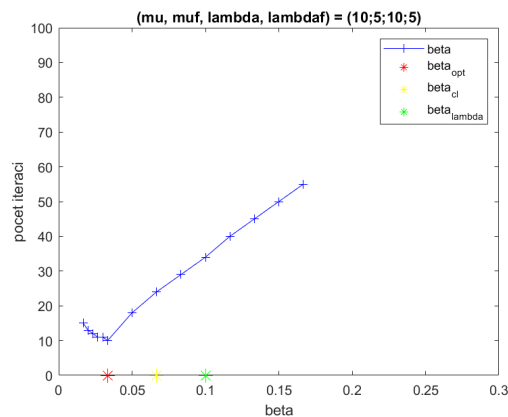
Pokud zmenšíme oba parametry v puklině, stane se optimálním parametrem β_{opt} s násobkem $c = \frac{9}{10}$, viz graf 3.14.



Graf. 3.15: Vliv prvního a druhého Lamého parametru v puklině na počet iterací, $(\lambda_f, \mu_f) = (0, 1; 0, 05)$

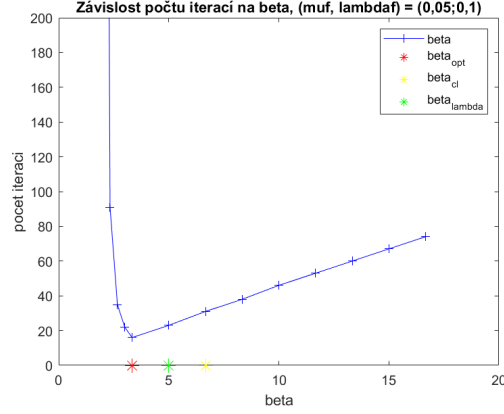
I při desetinásobném zmenšení oproti původním parametrům je optimálním násobkem $c = 1$, viz graf 3.15.

Nyní se tedy podíváme na to, jak se parametry chovají, pokud se mění oba dva Lamého parametry současně.



Graf. 3.16: Vliv prvního a druhého Lamého parametru na počet iterací, $(\lambda, \lambda_f, \mu, \mu_f) = (10; 5; 10; 5)$

V případě zvětšení obou dvou Lamého parametrů se celkově zmenší počet iterací potřebných k dosažení dané přesnosti, viz graf 3.16. Násobek optimálního parametru je nyní $c = 1$.



Graf. 3.17: Vliv prvního a druhého Lamého parametru v puklině na počet iterací, $(\lambda, \lambda_f, \mu, \mu_f) = (0.05; 0, 1)$

Z grafu 3.17 vidíme, že ani při zmenšení obou dvou Lamého parametrů se nezmění optimální násobek, který zůstává $c = 1$. Z grafů 3.16 a 3.17 vidíme, že pokud měníme oba Lamého parametry najednou, relaxační parametr se chová více stabilně a tak se pro oba případy dá použít násobek parametru $c = 1$. Pokud se změní parametr κ , neboli hydraulická vodivost, nemělo by to mít na relaxační parametr β přímý důsledek, a tím pádem by měl počet iterací při různých hodnotách κ zůstat konstantní.

Tabulka 3.5: Vliv konduktivity na počet iterací

	(κ, κ_f)		
	(1; 10)	(0,01; 0,1)	(0,1; 1)
β_λ	25	34	32
β_{cl}	25	34	32
β_{opt}	12	19	15

Můžeme vidět, že počet iterací zůstává přibližně stejný, i když malý vliv hydraulická konduktivita na počet iterací má. V počtu iterací se to ale nijak zásadně neprojeví, obzvlášť u parametru β_{opt} , viz tabulku 3.6.

V poslední řadě jsme provedli výpočty na pseudo-reálných parametrech a srovnali tak jak monolitické řešení (LU metoda), tak iterační (fixed stress splitting). Při testech byla postupně měněna velikost sítě a zachována přesnost $eps = 10^{-6}$.

Tabulka 3.6: Srovnání LU metody a metody fixed stress-splitting na pseudoreálných parametrech v závislosti na změně velikosti sítě.

	monoliticky		iteračně		
	LU		fixed stress splitting		
nx	t[s]	sm. odchylka	t[s]	sm. odchylka	n ite
16	5,79	1,87	6,87	2,27	3
32	33,25	6,21	29,60	4,33	3
64	X	X	92,33	11,02	3

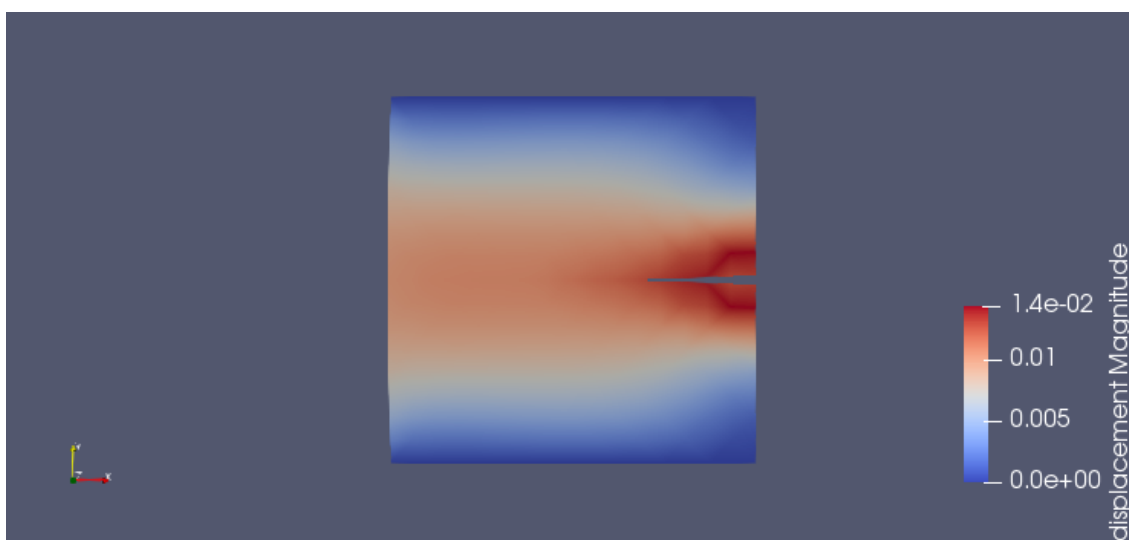
Z tabulky 3.6 vidíme, že při $nx = 16$ je rychlejší LU metoda, ale při zvětšení sítě se už jako časově úspornější řešič ukazuje iterační metoda fixed stress-splitting. Obě metody byly proměřeny v deseti časových krocích a z těchto deseti kroků byl vypočítán průměr a směrodatná odchylka. Vidíme, že směrodatná odchylka je mnohem větší, než při použití zkušebních parametrů: časová náročnost řešení jednoho časového kroku je tedy nepříliš stabilní. Při $nx = 64$, tedy při velikosti sítě $64 \times 64 \times 2$ elementů LU metoda nezkonvergovala z důvodu nedostatku paměti, což je jeden z hlavních nevýhod LU metody. Metoda fixed stress-splitting při $eps = 10^{-6}$ zkonvergovala po třech iteracích nezávisle na velikosti sítě. Pro metodu fixed stress-splitting byl použitý relaxační parametr $1 \cdot \beta_{opt}$, $1 \cdot \beta_{f_{opt}}$. Testy byly prováděny pro tyto pseudo-reálné parametry:

Tabulka 3.7: Pseudo-reálné parametry v jednotkách SI.

Parametr	Hodnota
T	1E-2
δ	1E-2
k	1E-6
k_f	1
S	1E-2
S_f	1E-1
f	(0;0)
f_f	(0;0)
μ	2E10
μ_f	2E6
λ	1E10
λ_f	1E6
α	1
α_f	1



Obrázek 3.4: Vizualizace tlaku v doméně v konečném čase.



Obrázek 3.5: Vizualizace vektoru posunutí v doméně v konečném čase.

Obrázky 3.4 a 3.5 byly vloženy pro lepší pochopení a vizualizaci problému. Toto řešení bylo spočteno na výpočetních parametrech.

4 Závěr

V této práci byly vyzkoušeny monolitické metody řešení Biotova modelu a následně byly porovnány na základě časové náročnosti. Na již existující kód byla aplikována metoda fixed stress-splitting a provedena analýza konvergence v závislosti na parametrech modelu.

Po všech provedených testech usuzujeme, že nejlepším parametrem je parametr β_{opt} , ale neexistuje násobek relaxačního parametru, který by obecně vyhovoval všem případům a parametrům modelu. Nejčastěji objevené hodnoty násobku c byly $c = 1$, $c = 3/2$, $c = 9/10$ a $c = 7/10$, tento násobek se mění při změně velikosti obou Lamého parametrů, jak v doméně tak v puklině, pokud další parametry zůstanou nezměněny. Pokud měníme pouze podíl mezi prvním nebo druhým Lamého parametrem v doméně a v puklině, nemá to na počet iterací vliv.

Výhodou iteračního řešení je efektivnější předpodmínění úloh proudění/mechaniky oproti monolitickému řešení, kde musíme používat robustní metody určené pro obecné regulární matice. Velkou výhodou oproti monolitickému řešení je v tomto případě možnost spočtení řešení i pro větší sítě. Další výhodou iteračního schématu je volba přesnosti (v textu zmiňováno jako *eps*), pokud víme, že nám postačuje určitá přesnost a tudíž ušetříme čas oproti monolitickému řešení a také možnost propojení s různými softwary pro proudění a mechaniku.

Nevýhodou iteračního řešení je vysoká citlivost relaxačního parametru na fyzikální parametry modelu. Nevhodné zvolení relaxačního parametru může prodloužit dobu řešení a zvýšit počet iterací.

Práce navazující na tuto by se mohly zabývat vlivem různých předpomiňovačů na časovou náročnost řešení při použití monolitického řešení, podrobnějšímu zkoumání parametru α na počet iterací metody fixed-stress splitting a srovnání iteračního a monolitického řešení při změně fyzikálních parametrů, tak jako bylo ukázáno v této práci při analýze konvergence.

Literatura

- [1] H.-D. CHEN, A. *Poroelasticity*. Springer, 2016. ISBN 978-3-319-25200-1.
- [2] ŠAUŠOVÁ, S. *Numerické modelování hydromechaniky v porézním prostředí*. Liberec, 2018. Diplomová práce. Technická univerzita v Liberci. Vedoucí práce Mgr. Jan Stebel, Ph.D.
- [3] 3 Hot dry rock (HDR) fields. <http://www.open.edu/openlearn/science-mathstechnology/science/environmental-science/energy-resources-geothermalenergy/content-section-3>. Accessed: 1999.
- [4] BIOT, M. A. General theory of three-dimensional consolidation. *Journal of applied physics* 12, 2 (1941), s. 155–164.
- [5] BIOT, M. A. Theory of elasticity and consolidation for a porous anisotropic solid. *Journal of applied physics* 26, 2 (1955), s. 182–185.
- [6] BIOT, M. A. General solutions of the equations of elasticity and consolidation for a porous material. *Journal of Applied Mechanics* 23, 1 (1956), s. 91–96.
- [7] BIOT, M. A. Nonlinear and semilinear rheology of porous solids. *Journal of Geophysical Research* 78, 23 (1973), s. 4924–4937.
- [8] BOTH, J. W., BORREGALES, M., NORDBOTTEN, J. M., KUMAR, K. and RADU, F. A. Robust fixed stress splitting for Biot’s equations in heterogeneous media. *Science Direct* [online], 2017. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0893965917300034>
- [9] LANGTANGEN, H. P. a LOGG, A. *Solving PDEs in Python – The FEniCS Tutorial Volume I*. [online], 18. září, 2017, s. 121. Dostupné z: <https://fenicsproject.org/pub/tutorial/pdf/fenics-tutorial-vol1.pdf>

- [10] TEBBENS, J. D., HNĚTYNKOVÁ, I., PLEŠINGER, M., STRAKOŠ, Z. a TICHÝ, P. *Analýza metod pro maticové výpočty: základní metody*. Praha: Mat-Fyzpres, 2012. ISBN 978-80-7378-201-6. Kolektivní monografie. Univerzita Karlova.
- [11] MIKELÍČ, A. a M. F. WHEELER. Convergence of iterative coupling for coupledflow and geomechanics. *Computational Geosciences*. 2013, 17(3), 455-461. DOI: 0.1007/s10596-012-9318-y.